Sierra Nevada Corporation Binary Armor SCADA Network Guard (NDcPP20E) Security Target

Version 0.7
July 31, 2018

Prepared for:

Sierra Nevada Corporation

11551 East Arapahoe Road
Centennial CO 80112

Prepared By:

Gossamer Laboratories

www.gossamersec.com

LIST OF TABLES

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is *Binary Armor*™ SCADA Network Guard provided by Sierra Nevada Corporation. The TOE is being evaluated as a Network Device.

The Security Target contains the following additional sections:

- Conformance Claims (Section 2)
- Security Objectives (Section 3)
- Extended Components Definition (Section 4)
- Security Requirements (Section 5)
- TOE Summary Specification (Section 6)

Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a parenthetical number placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***/selected-assignment/***]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.1 Security Target Reference

ST Title – Sierra Nevada Corporation Binary Armor SCADA Network Guard (NDcPP20E) Security Target

ST Version – Version 0.7

ST Date – July 31, 2018

1.2 TOE Reference

TOE Identification – Sierra Nevada Corporation Binary Armor SCADA Network Guard including the following hardware and software:

- Binary Armor hardware version 7000-SNC-01
- Binary Armor firmware version 1.6.19

- Binary Armor software suite version 1.6.19 consisting of:
  - Binary Armor Forge (management client that provides administrative access to the TOE)
  - (optional) Binary Armor Monitor (status and monitoring client for a single TOE)
  - (optional) Binary Armor Armory Client & Server (the Armory Server gathers status and monitoring from multiple TOEs and then makes that information available to Armory Clients (which connect to the Armory Server)).

TOE Developer – Sierra Nevada Corporation

Evaluation Sponsor – Sierra Nevada Corporation

1.3 TOE Overview

The Target of Evaluation (TOE) is Binary Armor (BA) SCADA Network Guard. The Binary Armor® SCADA Network Guard provides critical, real-time, endpoint cybersecurity for Supervisory Control and Data Acquisition (SCADA) network systems.

1.4 TOE Description

BA is designed for in-line installation between Programmable Logic Controllers (PLCs), remote terminal units, intelligent electronic devices or controllers and the WAN/LAN, to provide bi-directional security across all communication layers. Binary Armor supports TLS encryption. The TOE provides two, separate, physical interfaces: a high NIC (typically connected to SCADA/ICS equipment) and a low NIC-(typically connected to external systems such as Human Machine Interface, HMI). The TOE supports remote administration over the network as well as local administration (through a directly networked workstation).

The TOE works by processing every byte of every message with a dynamic state-based rule-set that processes messages based on system control logic. This process ensures only safe message traffic reaches critical SCADA systems.

For the purpose of this evaluation, BA will be treated as a network device offering CAVP certified cryptographic functions, security auditing, secure administration, trusted updates, self-tests, and secure connections to other servers (e.g., to transmit audit records).

1.4.1 TOE Architecture

BA is provided as a hardware network appliance. The product provides an administrative interface over TLS.

1.4.1.1 Physical Boundaries

The TOE has a rugged enclosure that protects it from modification and contains a single embedded board containing an Intel Atom E3845 processor, memory, and flash storage. The TOE hardware contains a hardened operating system (RHEL 7.4) that does not permit operators (even an authorized administrator) access to the OS with SNC developed firmware running atop. The TOE provides a TLS-protected management interface which is accessed via SNC's Forge, Armory, and Monitor applications which run on a PC/workstation which is part of the operating environment of the TOE.[1] An administrator can configure the TOE for remote access on either its high or low network interface. The administrator always accesses the TOE through its TLS management interface, irrespective of whether the administrator configured the TOE to listen for management connections on its low or high network interface and irrespective of whether the administrator accesses the TOE remotely or locally.

The administrator gains local access by physically pressing and holding the TOE's configuration (CFG) button and then accessing the TOE's TLS management interface from a directly networked workstation. In this context,

[1] The PC/workstation is part of the operating environment of the TOE.

"directly networked" means connected via "crossover" cable or through a network switch to which only the TOE and the workstation are connected.

The TOE can be configured to forward its audit records to an external syslog server in the network environment. This is generally advisable given the audit log storage space on the evaluated appliance.

Additionally, an administrator can optionally configure the TOE to solicit time from an NTP server, if present in the environment (the administrator can manually set the TOE's time if no such NTP server exists).

The TOE's Operating Environment includes the following:

1) A Windows workstation - The Binary Armor software suite of tools operates on Microsoft Windows 7, 8 or 10.
2) A security token in the form of a PKCS#11 compliant smart card or USB device present on the workstation. The security token is used by the TOE to sign and encrypt configuration files and to activate override mode (described in Section 6.3).[2] The token is configured by loading private/public key pairs in the form of X509 certificates onto the TOE and then pairing them to the override and configuration operations on the device.
3) A TLS-protected syslog server that receives audit events from the TOE
4) An NTP server with which the TOE can synchronize its clock

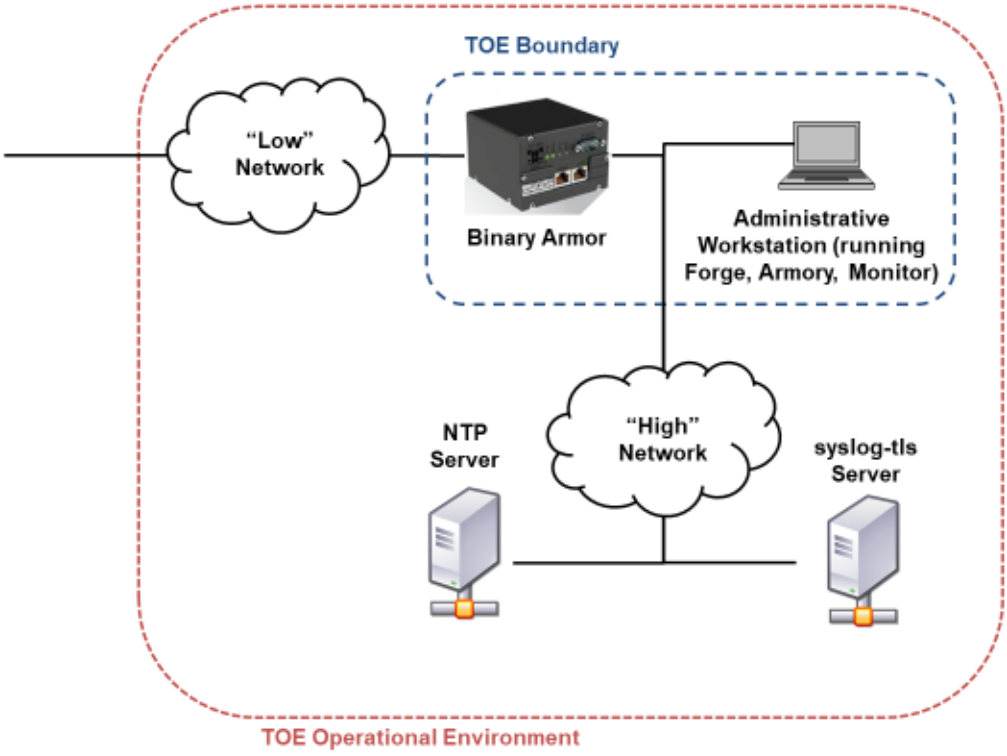

Figure 1: TOE Boundaries

[2] The security token related functions (pairing, signing, encrypting and activating) have not been evaluated and are outside the scope of this evaluation.

1.4.1.2 Logical Boundaries

This section summarizes the security functions provided by Binary Armor:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

1.4.1.2.1 Security audit

The TOE generates audit events associated with identification and authentication, management, updates, and user sessions. The TOE can store the events in a local log or export them to a syslog server using a TLS protected channel.

1.4.1.2.2 Cryptographic support

The TOE provides CAVP certified cryptography in support of its TLS implementation and administrator authentication. Cryptographic services include key management, random bit generation, encryption/decryption, digital signature and secure hashing.

1.4.1.2.3 Identification and authentication

The TOE requires users to be identified and authenticated before they can use functions mediated by the TOE, with the exception of reading the login banner, obtaining status, and (if configured) restarting the TOE and enabling override (see the TSS for a description of the TOE's override mode). It provides the ability to both assign attributes (user password) and to authenticate users against these attributes. The TOE also provides X.509 certificate checking for its TLS connections.

1.4.1.2.4 Security management

The TOE provides a management interface that an administrator can access via a network port. The SNC Forge, Monitor, and Armory applications utilize the TOE's API. The management interface is protected with TLS. The management interface is limited to the authorized administrator.

1.4.1.2.5 Protection of the TSF

The TOE provides a variety of means of protecting itself. The TOE performs self-tests that cover the correct operation of the TOE. It provides functions necessary to securely update the TOE. It relies upon either manually provided time or an NTP server in its environment to ensure reliable timestamps. It protects sensitive data such as passwords and cryptographic keys stored on the TOE's internal Flash so that they are not accessible even by an authorized administrator.

1.4.1.2.6 TOE access

The TOE can be configured to display a logon banner before a user session is established. The TOE also enforces inactivity timeouts for administrative sessions.

1.4.1.2.7 Trusted path/channels

The TOE provides local administration which is subject to physical protection. To access local administrator, an operator must directly network their workstation to the TOE (again, e.g., a "crossover" cable or through a network switch to which only the TOE and the workstation are connected), and then must physically press the TOE's configuration button. This transitions the TOE into its configuration mode, where an administrator can locally

configure it. For both local and remote access, the administrative session is protected by TLS thus ensuring protection against modification and disclosure.

The TOE also protects its audit records from modification and disclosure by using TLS to communicate with the syslog server.

1.4.2 TOE Documentation

The following list of documents was examined as part of the evaluation:

- Administrator Guide for Common Criteria for Binary Armor, 0318-0200-0001, Rev B, 3 Aug 18 (**Admin Guide**)
- Binary Armor User Manual, 0318-0100-0015, Rev B, 17 July 18 (**User Manual**)

2. Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 4, September 2012.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 4, September 2012.
    - Part 3 Conformant
- Package Claims:
    - collaborative Protection Profile for Network Devices, Version 2.0 + Errata 20180314, Version 2.0, 14 March 2018 (**NDcPP20EE**)
- Technical Decisions:
    - Applicable NIAP Technical decisions as of June 8, 2018: TD0228, TD0256, TD0257, TD0259, TD0260, TD0262, TD0281, TD0289, TD0290, TD0291, TD0321, TD0322, TD0323, TD0324.

TD No.	Applied?	Rationale
TD0291	Yes	
TD0290	Yes	
TD0289	Yes	
TD0281	No	SSH not claimed
TD0262	No	TLSS mutual authentication not selected (FCS_TLSS_EXT.2 not present))
TD0260	No	SSH not claimed
TD0259	No	SSH not claimed
TD0257	Yes	
TD0256	No	TLSC mutual authentication not selected (FCS_TLSC_EXT.2 not present)
TD0228	Yes	
TD0321	No	Effective date is July 1, 2018
TD0322	No	FCS_TLSS_EXT.2 not claimed
TD0323	No	FCS_DTLSS_EXT.2 not claimed
TD0324	No	This TD corrects section numbers in the SD

Table 1: Technical Decisions

2.1 Conformance Rationale

The ST conforms to the NDcPP20E. As explained previously, the security problem definition, security objectives, and security requirements have been drawn from the PP.

3. Security Objectives

The Security Problem Definition may be found in the NDcPP20E and this section reproduces only the corresponding Security Objectives for operational environment for reader convenience. The NDcPP20E offers additional information about the identified security objectives, but that has not been reproduced here and the NDcPP20E should be consulted if there is interest in that material.

In general, the NDcPP20E has defined Security Objectives appropriate for Network Devices and as such are applicable to the Binary Armor SCADA Network Guard TOE.

3.1 Security Objectives for the Operational Environment

OE.ADMIN_CREDENTIALS_SECURE The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

OE.NO_GENERAL_PURPOSE There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.

OE.NO_THRU_TRAFFIC_PROTECTION The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.

OE.PHYSICAL Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.

OE.RESIDUAL_INFORMATION The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

OE.TRUSTED_ADMIN TOE Administrators are trusted to follow and apply all guidance documentation in a trusted manner.

OE.UPDATES The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.

T.SECURITY_FUNCTIONALITY_FAILURE An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

4. Extended Components Definition

All of the extended requirements in this ST have been drawn from the NDcPP20E. The NDcPP20E defines the following extended requirements and since they are not redefined in this ST the NDcPP20E should be consulted for more information in regard to those CC extensions.

Extended SFRs:

- FAU_STG_EXT.1: Protected Audit Event Storage
- FCS_RBG_EXT.1: Random Bit Generation
- FCS_TLSC_EXT.1: TLS Client Protocol
- FCS_TLSS_EXT.1: TLS Server Protocol
- FIA_PMG_EXT.1: Password Management
- FIA_UAU_EXT.2: Password-based Authentication Mechanism
- FIA_UIA_EXT.1: User Identification and Authentication
- FIA_X509_EXT.1(2): X.509 Certificate Validation (Rev)
- FIA_X509_EXT.2: X.509 Certificate Authentication
- FIA_X509_EXT.3: X.509 Certificate Requests
- FPT_APW_EXT.1: Protection of Administrator Passwords
- FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
- FPT_STM_EXT.1: Reliable Time Stamps
- FPT_TST_EXT.1: TSF testing
- FPT_TUD_EXT.1: Trusted update
- FTA_SSL_EXT.1: TSF-initiated Session Locking

5. Security Requirements

This section defines the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that serve to represent the security functional claims for the Target of Evaluation (TOE) and to scope the evaluation effort.

The SFRs and SARs have all been drawn from the NDcPP20E. The refinements and operations already performed in the NDcPP20E are not identified (e.g., highlighted) here, rather the requirements have been copied from the NDcPP20E and any residual operations have been completed herein. Of particular note, the NDcPP20E made a number of refinements and completed some of the SFR operations defined in the Common Criteria (CC) and that PP should be consulted to identify those changes if necessary.

5.1 TOE Security Functional Requirements

The following table identifies the SFRs that are satisfied by Binary Armor SCADA Network Guard TOE.

Requirement Class	Requirement Component
FAU: Security audit	FAU_GEN.1: Audit Data Generation
	FAU_GEN.2: User identity association
	FAU_STG_EXT.1: Protected Audit Event Storage
FCS: Cryptographic	FCS_CKM.1: Cryptographic Key Generation
support	FCS_CKM.2: Cryptographic Key Establishment
	FCS_CKM.4: Cryptographic Key Destruction
	FCS_COP.1/DataEncryption: Cryptographic Operation (AES Data Encryption/Decryption)
	FCS_COP.1/SigGen: Cryptographic Operation (Signature Generation and Verification)
	FCS_COP.1/Hash: Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash: Cryptographic Operation (Keyed Hash Algorithm)
	FCS_RBG_EXT.1: Random Bit Generation
	FCS_TLSC_EXT.1: TLS Client Protocol
	FCS_TLSS_EXT.1: TLS Server Protocol
FIA: Identification and	FIA_AFL.1: Authentication Failure Management
authentication	FIA_PMG_EXT.1: Password Management
	FIA_UAU.7: Protected Authentication Feedback
	FIA_UAU_EXT.2: Password-based Authentication Mechanism
	FIA_UIA_EXT.1: User Identification and Authentication
	FIA_X509_EXT.1/Rev: X.509 Certificate Validation (Rev)
	FIA_X509_EXT.2: X.509 Certificate Authentication
	FIA_X509_EXT.3: X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate: Management of security functions behaviour
	FMT_MOF.1Functions: Management of security functions behaviour
	FMT_MTD.1/CoreData: Management of TSF Data
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.2: Restrictions on Security Roles
FPT: Protection of the TSF	FPT_APW_EXT.1: Protection of Administrator Passwords
	FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)
	FPT_STM_EXT.1: Reliable Time Stamps

	FPT_TST_EXT.1: TSF testing
	FPT_TUD_EXT.1: Trusted update
FTA: TOE access	FTA_SSL.3: TSF-initiated Termination
	FTA_SSL.4: User-initiated Termination
	FTA_SSL_EXT.1: TSF-initiated Session Locking
	FTA_TAB.1: Default TOE Access Banners
FTP: Trusted	FTP_ITC.1: Inter-TSF trusted channel
path/channels	FTP_TRP.1/Admin: Trusted Path (Admin)

Table 2: TOE Security Functional Components

5.1.1 Security audit (FAU)

5.1.1.1 Audit Data Generation (FAU_GEN.1)

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:
a) Start-up and shut-down of the audit functions;
b) All auditable events for the not specified level of audit; and
c) All administrative actions comprising:
- Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).
- Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).
- Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).
- Resetting passwords (name of related user account shall be logged).
- [***no other actions***];

d) Specifically defined auditable events listed in the Table below.

Requirement	Auditable Events	Additional Content
FAU_GEN.1	None	None
FAU_GEN.2	None	None
FAU_STG_EXT.1	None	None
FCS_CKM.1	None	None
FCS_CKM.2	None	None
FCS_CKM.4	None	None
FCS_COP.1/DataEncryption	None	None
FCS_COP.1/SigGen	None	None
FCS_COP.1/Hash	None	None
FCS_COP.1/KeyedHash	None	None
FCS_RBG_EXT.1	None	None
FCS_TLSC_EXT.1	Failure to establish a TLS Session.	Reason for failure.
FCS_TLSS_EXT.1	Failure to establish a TLS Session.	Reason for failure.
FIA_AFL.1	Unsuccessful login attempt limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None	None
FIA_UAU.7	None	None
FIA_UAU_EXT.2	All use of identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UIA_EXT.1	All use of identification and	Origin of the attempt (e.g., IP address).

	authentication mechanism.	
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate.	Reason for failure.
FIA_X509_EXT.2	None	None
FIA_X509_EXT.3	None	None
FMT_MOF.1/ManualUpdate	Any attempt to initiate a manual update.	None
FMT_MOF.1/Functions	Modification of the behavior of the transmission of audit data to an external IT entity, the handling of audit data, the audit functionality when Local Audit Storage Space is full.	None
FMT_MTD.1/CoreData	All management activities of TSF data.	None
FMT_SMF.1	None	None
FMT_SMR.2	None	None
FPT_APW_EXT.1	None	None
FPT_SKP_EXT.1	None	None
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	For discontinuous changes to time: The old and new values for the time. Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TST_EXT.1	None	None
FPT_TUD_EXT.1	Initiation of update; result of the update attempt (success or failure).	None
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None
FTA_SSL.4	The termination of an interactive session.	None
FTA_SSL_EXT.1	(if 'lock the session' is selected) Any attempts at unlocking of an interactive session. (if 'terminate the session' is selected) The termination of a local session by the session locking mechanism.	None
FTA_TAB.1	None	None
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None

Table 3: Audit Table

FAU_GEN.1.2

The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, information specified in column three of Table 2.

5.1.1.2 User identity association (FAU_GEN.2)

FAU_GEN.2.1

For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.1.1.3 Protected Audit Event Storage (FAU_STG_EXT.1)

FAU_STG_EXT.1.1

The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

FAU_STG_EXT.1.2

The TSF shall be able to store generated audit data on the TOE itself.

FAU_STG_EXT.1.3

The TSF shall [***overwrite previous audit records according to the following rule: [overwriting the oldest log record first]***] when the local storage space for audit data is full.

5.1.2 Cryptographic support (FCS)

5.1.2.1 Cryptographic Key Generation (FCS_CKM.1)

FCS_CKM.1.1

The TSF shall generate asymmetric cryptographic keys in accordance with a specified cryptographic key generation algorithm: [

- ***RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.3,***
- ***ECC schemes using 'NIST curves' [P-256, P-384] that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.4,***
- ***FFC schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Appendix B.1***].

(TD0291 Applied)

5.1.2.2 Cryptographic Key Establishment (FCS_CKM.2)

FCS_CKM.2.1

The TSF shall perform cryptographic key establishment in accordance with a specified cryptographic key establishment method: [

- ***RSA-based key establishment schemes that meet the following: NIST Special Publication 800-56B Revision 1, 'Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography',***
- ***Elliptic curve-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography',***
- ***Finite field-based key establishment schemes that meet the following: NIST Special Publication 800-56A Revision 2, 'Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography'***].

5.1.2.3 Cryptographic Key Destruction (FCS_CKM.4)

FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [

- ***For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes]];***
- ***For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [o logically addresses the storage location of the key and performs a [single] overwrite consisting of [zeroes]]***]

that meets the following: No Standard.

5.1.2.4 Cryptographic Operation (AES Data Encryption/Decryption) (FCS_COP.1/DataEncryption)

FCS_COP.1.1/DataEncryption

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithm AES used in [***CBC, GCM***] mode and cryptographic key sizes [***128 bits, 256 bits***] that meet the following: AES as specified in ISO 18033-3, [***CBC as specified in ISO 10116, GCM as specified in ISO 19772***].

5.1.2.5 Cryptographic Operation (Hash Algorithm) (FCS_COP.1/Hash)

FCS_COP.1.1/Hash

The TSF shall perform cryptographic hashing services in accordance with a specified cryptographic algorithm [***SHA-256, SHA-384, SHA-512***] and message digest sizes [***256, 384, 512***] that meet the following: ISO/IEC 10118-3:2004.

5.1.2.6 Cryptographic Operation (Keyed Hash Algorithm) (FCS_COP.1/KeyedHash)

FCS_COP.1.1/KeyedHash

The TSF shall perform keyed-hash message authentication in accordance with a specified cryptographic algorithm [***HMAC-SHA-256, HMAC-SHA-384***] and cryptographic key sizes [***256, 384***] and message digest sizes [***256, 384***] bits that meet the following: ISO/IEC 9797-2:2011, Section 7 'MAC Algorithm 2'.

5.1.2.7 Cryptographic Operation (Signature Generation and Verification) (FCS_COP.1/SigGen)

FCS_COP.1.1/SigGen

The TSF shall perform cryptographic signature services (generation and verification) in accordance with a specified cryptographic algorithm [
- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048, 3072],
- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256, 384, 521]]
that meet the following:
[***- For RSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSA-PKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,***
- For ECDSA schemes: FIPS PUB 186-4, 'Digital Signature Standard (DSS)', Section 6 and Appendix D, Implementing 'NIST curves' [P-256, P-384, P-521]; ISO/IEC 14888-3, Section 6.4].

5.1.2.8 Random Bit Generation (FCS_RBG_EXT.1)

FCS_RBG_EXT.1.1

The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [***CTR_DRBG (AES)***].

FCS_RBG_EXT.1.2

The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [***[one] hardware-based noise source***] with a minimum of [***256 bits***] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and CSPs that it will generate.

5.1.2.9 TLS Client Protocol (FCS_TLSC_EXT.1)

FCS_TLSC_EXT.1.1

The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:

[***TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,***
TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289].

FCS_TLSC_EXT.1.2

The TSF shall verify that the presented identifier matches the reference identifier per RFC 6125 section 6.

FCS_TLSC_EXT.1.3

The TSF shall only establish a trusted channel if the server certificate is valid. If the server certificate is deemed invalid, then the TSF shall [***not establish the connection***].

FCS_TLSC_EXT.1.4

The TSF shall [***present the Supported Elliptic Curves Extension with the following NIST curves: [secp256r1, secp384r1, secp521r1] and no other curves***] in the Client Hello.

5.1.2.10 TLS Server Protocol (FCS_TLSS_EXT.1)

FCS_TLSS_EXT.1.1

The TSF shall implement [***TLS 1.2 (RFC 5246), TLS 1.1 (RFC 4346)***] and reject all other TLS and SSL versions. The TLS implementation will support the following ciphersuites:
[***TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,***
TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 4492,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 4492,
TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246,
TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289].

FCS_TLSS_EXT.1.2

The TSF shall deny connections from clients requesting SSL 2.0, SSL 3.0, TLS 1.0, and [***none***].

FCS_TLSS_EXT.1.3

The TSF shall [***perform RSA key establishment with key size [2048 bits, 3072 bits], generate EC***

Diffie-Hellman parameters over NIST curves [secp256r1] and no other curves, generate Diffie-Hellman parameters of size [2048 bits]].

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication Failure Management (FIA_AFL.1)

FIA_AFL.1.1

The TSF shall detect when an Administrator configurable positive integer within [***3-10***] unsuccessful authentication attempts occur related to Administrators attempting to authenticate remotely.

FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met, the TSF shall [***prevent the offending remote Administrator from successfully authenticating until an Administrator defined time period has elapsed***].

5.1.3.2 Password Management (FIA_PMG_EXT.1)

FIA_PMG_EXT.1.1

The TSF shall provide the following password management capabilities for administrative passwords:

a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [***'!', '@', '#', '$', '%', '^', '&', '*', '(', ')', [no other characters]***];

b) Minimum password length shall be configurable to [***4***] and [***15***].

5.1.3.3 Protected Authentication Feedback (FIA_UAU.7)

FIA_UAU.7.1

The TSF shall provide only obscured feedback to the administrative user while the authentication is in progress at the local console.

5.1.3.4 Password-based Authentication Mechanism (FIA_UAU_EXT.2)

FIA_UAU_EXT.2.1

The TSF shall provide a local password-based authentication mechanism, and [***[no other authentication mechanism]***] to perform local administrative user authentication.

5.1.3.5 User Identification and Authentication (FIA_UIA_EXT.1)

FIA_UIA_EXT.1.1

The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- [***obtain status information***
- request TOE certificate
- restart the TOE[3]
- enable override
- acknowledge alarm].

FIA_UIA_EXT.1.2

The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of that administrative user.

[3] An administrator can configure whether the TOE allows restart and override

5.1.3.6 X.509 Certificate Validation (Rev) (FIA_X509_EXT.1/Rev)

FIA_X509_EXT.1.1/Rev

The TSF shall validate certificates in accordance with the following rules:
- RFC 5280 certificate validation and certificate path validation supporting a minimum path length of three certificates.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certificate path by ensuring the presence of the basicConstraints extension and that the CA flag is set to TRUE for all CA certificates.
- The TSF shall validate the revocation status of the certificate using [***a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3, Certificate Revocation List (CRL) as specified in RFC 5759 Section 5***]
- The TSF shall validate the extendedKeyUsage field according to the following rules:

o Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.

o Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.

o Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.

o OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.

FIA_X509_EXT.1.2/Rev

The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.1.3.7 X.509 Certificate Authentication (FIA_X509_EXT.2)

FIA_X509_EXT.2.1

The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [***TLS***], and [***no additional uses***].

FIA_X509_EXT.2.2

When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [***accept the certificate***].

5.1.3.8 X.509 Certificate Requests (FIA_X509_EXT.3)

FIA_X509_EXT.3.1

The TSF shall generate a Certificate Request Message as specified by RFC 2986 and be able to provide the following information in the request: public key and [***Common Name***].

FIA_X509_EXT.3.2

The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behaviour (Manual Update) (FMT_MOF.1/ManualUpdate)

FMT_MOF.1.1/ManualUpdate

The TSF shall restrict the ability to enable the functions to perform manual update to Security Administrators.

5.1.4.2 Management of security functions behaviour (Functions) (FMT_MOF.1/Functions)

FMT_MOF.1.1/Functions

The TSF shall restrict the ability to [***modify the behaviour of***] the functions [***transmission of audit data to an external IT entity***] to Security Administrators.

5.1.4.3 Management of TSF Data (CoreData) (FMT_MTD.1/CoreData)

FMT_MTD.1.1/CoreData

The TSF shall restrict the ability to manage the TSF data to Security Administrators.

5.1.4.4 Specification of Management Functions (FMT_SMF.1)

FMT_SMF.1.1

The TSF shall be capable of performing the following management functions:
- Ability to administer the TOE locally and remotely;
- Ability to configure the access banner;
- Ability to configure the session inactivity time before session termination or locking;
- Ability to update the TOE, and to verify the updates using [***digital signature***] capability prior to installing those updates;
- Ability to configure the authentication failure parameters for FIA_AFL.1;
- [***o Ability to configure audit behavior***].

5.1.4.5 Restrictions on Security Roles (FMT_SMR.2)

FMT_SMR.2.1

The TSF shall maintain the roles: - Security Administrator.

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions
- The Security Administrator role shall be able to administer the TOE locally;
- The Security Administrator role shall be able to administer the TOE remotely

are satisfied.

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Protection of Administrator Passwords (FPT_APW_EXT.1)

FPT_APW_EXT.1.1

The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2

The TSF shall prevent the reading of plaintext passwords.

5.1.5.2 Protection of TSF Data (for reading of all pre-shared, symmetric and private keys) (FPT_SKP_EXT.1)

FPT_SKP_EXT.1.1

The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.1.5.3 Reliable Time Stamps (FPT_STM_EXT.1)

FPT_STM_EXT.1.1

The TSF shall be able to provide reliable time stamps for its own use.

FPT_STM_EXT.1.2

The TSF shall [***allow the Security Administrator to set the time, synchronise time with external time sources***].

5.1.5.4 TSF testing (FPT_TST_EXT.1)

FPT_TST_EXT.1.1

The TSF shall run a suite of the following self-tests [***during initial start-up (on power on)***] to demonstrate the correct operation of the TSF: [***OpenSSL performs self-test including the***

following cryptographic algorithm Known Answer Tests (KATs): AES, RSA, ECDSA, DH, ECDH, DRBG, HMAC, and SHA and an integrity test: HMAC-SHA-256, NSS performs the follows KATs: ECDSA, RSA, and SHA and a module integrity test: DSA 2048 w/ SHA-256].

5.1.5.5 Trusted update (FPT_TUD_EXT.1)

FPT_TUD_EXT.1.1

The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [***no other TOE firmware/software version***].

FPT_TUD_EXT.1.2

The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [***no other update mechanism***].

FPT_TUD_EXT.1.3

The TSF shall provide means to authenticate firmware/software updates to the TOE using a [***digital signature mechanism***] prior to installing those updates.

5.1.6 TOE access (FTA)

5.1.6.1 TSF-initiated Termination (FTA_SSL.3)

FTA_SSL.3.1

The TSF shall terminate a remote interactive session after a Security Administrator-configurable time interval of session inactivity.

5.1.6.2 User-initiated Termination (FTA_SSL.4)

FTA_SSL.4.1

The TSF shall allow Administrator-initiated termination of the Administrator's own interactive session.

5.1.6.3 TSF-initiated Session Locking (FTA_SSL_EXT.1)

FTA_SSL_EXT.1.1

The TSF shall, for local interactive sessions, [***- terminate the session***] after a Security Administrator-specified time period of inactivity.

5.1.6.4 Default TOE Access Banners (FTA_TAB.1)

FTA_TAB.1.1

Before establishing an administrative user session the TSF shall display a Security Administrator-specified advisory notice and consent warning message regarding use of the TOE.

5.1.7 Trusted path/channels (FTP)

5.1.7.1 Inter-TSF trusted channel (FTP_ITC.1)

FTP_ITC.1.1

The TSF shall be capable of using [***TLS***] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, [***no other capabilities***] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2

The TSF shall permit the TSF or the authorized IT entities to initiate communication via the trusted channel.

FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [***audit/syslog***].

5.1.7.2 Trusted Path (Admin) (FTP_TRP.1/Admin: Trusted Path)

FTP_TRP.1.1/Admin

The TSF shall be capable of using [***TLS***] to provide a communication path between itself and authorized remote Administrators that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from disclosure and provides detection of modification of the channel data.

FTP_TRP.1.2/Admin

The TSF shall permit remote Administrators to initiate communication via the trusted path.

FTP_TRP.1.3/Admin

The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.2 TOE Security Assurance Requirements

The SARs for the TOE are the components as specified in Part 3 of the Common Criteria. Note that the SARs have effectively been refined with the assurance activities explicitly defined in association with both the SFRs and SARs.

Requirement Class	Requirement Component
ADV: Development	ADV_FSP.1: Basic Functional Specification
AGD: Guidance documents	AGD_OPE.1: Operational User Guidance
	AGD_PRE.1: Preparative Procedures
ALC: Life-cycle support	ALC_CMC.1: Labelling of the TOE
	ALC_CMS.1: TOE CM Coverage
ATE: Tests	ATE_IND.1: Independent Testing Conformance
AVA: Vulnerability assessment	AVA_VAN.1: Vulnerability Survey

Table 4 Assurance Components

5.2.1 Development (ADV)

5.2.1.1 Basic Functional Specification (ADV_FSP.1)

ADV_FSP.1.1d

The developer shall provide a functional specification.

ADV_FSP.1.2d

The developer shall provide a tracing from the functional specification to the SFRs.

ADV_FSP.1.1c

The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2c

The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3c

The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4c

The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

ADV_FSP.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.2.2 Guidance documents (AGD)

5.2.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d

The developer shall provide operational user guidance.

AGD_OPE.1.1c

The operational user guidance shall describe, for each user role, the useraccessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2c

The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3c

The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4c

The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5c

The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6c

The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7c

The operational user guidance shall be clear and reasonable.

AGD_OPE.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d

The developer shall provide the TOE, including its preparative procedures.

AGD_PRE.1.1c

The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_PRE.1.2c

The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

AGD_PRE.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AGD_PRE.1.2e
The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.2.3 Life-cycle support (ALC)

5.2.3.1 Labelling of the TOE (ALC_CMC.1)

ALC_CMC.1.1d
The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.1.1c
The TOE shall be labelled with its unique reference.

ALC_CMC.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.3.2 TOE CM Coverage (ALC_CMS.1)

ALC_CMS.1.1d
The developer shall provide a configuration list for the TOE.

ALC_CMS.1.1c
The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.1.2c
The configuration list shall uniquely identify the configuration items.

ALC_CMS.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.2.4 Tests (ATE)

5.2.4.1 Independent Testing Conformance (ATE_IND.1)

ATE_IND.1.1d
The developer shall provide the TOE for testing.

ATE_IND.1.1c
The TOE shall be suitable for testing.

ATE_IND.1.1e
The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2e
The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.2.5 Vulnerability assessment (AVA)

5.2.5.1 Vulnerability Survey (AVA_VAN.1)

AVA_VAN.1.1d
The developer shall provide the TOE for testing.

AVA_VAN.1.1c
The TOE shall be suitable for testing.

AVA_VAN.1.1e

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence..

AVA_VAN.1.2e

The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3e

The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the security functions:

- Security audit
- Cryptographic support
- Identification and authentication
- Security management
- Protection of the TSF
- TOE access
- Trusted path/channels

6.1 Security audit

The Security audit function is designed to satisfy the following security functional requirements:

FAU_GEN.1/2: The TOE provides an audit log (the ba.log) and provides logging of the audit information described in the table in section 5.1.1.1. Additionally, the TOE records audit records that includes the action requested, the success or failure, and the source IP address when the administrator requests the TOE generate a self-signed certificate, generate/import a CSR, and imports an (off-device generated) certificate. When auditing certificate operations, the TOE records the certificate's Subject, Issuer, Valid dates, and Key Type, in order to uniquely identify the certificate.

FAU_STG_EXT.1: The TOE stores audit logs locally and provides the administrator the ability to configure the real-time export of syslog records protected with TLS. The TOE provides 100 megabytes of local storage for audit logs, and the TOE permits no access to the audit logs other than allowing an authenticated administrator to download a copy of the logs or to clear the logs. The TOE will delete the oldest audit records in order to free up space for new audit records when the TOE reaches the maximum local storage space for audit data. The TOE provides ten rotations and ensures that the oldest rotation is deleted.

6.2 Cryptographic support

The Cryptographic support function is designed, as described in the table below, to satisfy the security functional requirements.

The TOE includes the Red Hat OpenSSL and Red Hat NSS libraries. Each of these libraries possess CAVP certificates for their different cryptographic algorithms. Table 5 and 6 below summarize the CAVP certificates.

The TOE uses its Red Hat OpenSSL library for all TLS and certificate functionality.

SFR	Algorithm	NIST Standard	Cert#
FCS_CKM.1 (Key Gen)	RSA IFC Key Generation	FIPS 186-4, RSA	2976
	ECDSA ECC Key Generation	FIPS 186-4, ECDSA	1495
	DSA FFC Key Generation	FIPS 186-4, DSA	1425
FCS_CKM.2 (Key Establishment)	RSA-based Key Exchange	Vendor affirm 800-56B SHS DRBG RSA	N/A
	ECC-based Key Exchange	SP 800-56A, CVL KAS ECC	1986
	FFC-based Key Exchange	SP 800-56A, CVL KAS FFC	1986
FCS_COP.1/DataEncryption	AES 128/256 CBC, GCM	ISO 18033-3 (AES)	5544

SFR	Algorithm	NIST Standard	Cert#
(AES)		ISO 10116 (CBC) ISO 19772 (GCM)	
FCS_COP.1/SigGen	RSA Sign/Verify	FIPS 186-4, RSA	2976
	ECDSA Sign/Verify	FIPS 186-4, ECDSA	1495
FCS_COP.1/Hash	SHA Hashing	ISO/IEC 10118-3:2004	4450
FCS_COP.1/KeyedHash	HMAC-SHA	FIPS 198-1 & 180-4	3695
FCS_RBG_EXT.1 (Random)	DRBG Bit Generation	ISO/IEC 18031:2011	2199

Table 5 OpenSSL Cryptographic Algorithms

The TOE uses its Red Hat NSS library to verify the signatures of trusted updates (through RPM and the yum package manager).

SFR	Algorithm	NIST Standard	Cert#
FCS_COP.1/SigGen	RSA Sign/Verify	FIPS 186-4, RSA	2975
	ECDSA Sign/Verify	FIPS 186-4, ECDSA	1494
FCS_COP.1/Hash	SHA Hashing	ISO/IEC 10118-3:2004	4449

Table 6 NSS CAVP Certificates

FCS_CKM.1: The TOE supports generating keypairs both for authentication and for key exchange. When generating authentication keypairs, the TOE can generate a CSR with an RSA 2048 bit keypair and an ECDSA P-256 or P-384 keypair. For key establishment, the TOE will generate either 2048-bit DHE keys or P-256 ECDHE keys (depending on the ciphersuite strength) during TLS negotiation.

FCS_CKM.2: The TOE performs key establishment for TLS, and the TOE acts as both a TLS server (to service incoming administrative sessions) and as a TLS client (for syslog export).

FCS_CKM.4: The TOE clears keys from both volatile memory (by overwriting the memory locations in RAM with zeros) and from non-volatile (by first overwriting the contents of the file in the Flash filesystem with zeros, then sync'ing, and finally deleting the file).

CSP or Key:	Stored in	Zeroized upon:	Zeroized by:
TLS Host RSA or ECDSA private key	On Flash	Command	Overwriting with pseudo-random data
TLS pre-master secret	In Memory	Handshake done	Overwriting with zeros
TLS session key	In Memory	Close of session	Overwriting with zeros
Passwords	On Flash in a SHA-256 hash	Command	Overwriting once with pseudo-random data

Table 7: Key Clearing

FCS_COP.1/DataEncryption/Hash/KeyedHash/SigGen: The TOE uses most cryptographic algorithms in support of TLS. For example, the TOE uses SHA hashing both during digital signature calculation (hashing of the message) as well as for integrity as part of HMAC-SHA operations within TLS. The TOE uses AES-CBC and GCM (both 128 and 256-bit) depending upon the TLS cipher suite. Likewise, during TLS authentication, The TOE will generate RSA 2048/3072 or ECDSA P-256/P-384/P-521 signatures depending upon what certificate the administrator has configured. The TOE uses HMAC-SHA-256 and 384 for integrity of TLS protected data using SHA-256/384 with 256/384-bit keys to produce a 256/384 output MAC. The SHA-256 and 384 algorithms have block sizes of 512 and 1024-bits respectively. The TOE also uses cryptography when verifying the signatures of updated packages and will use either RSA or ECDSA verification (depending upon the signing key).

FCS_RBG_EXT.1: The TOE uses an AES-256 CTR_DRBG from OpenSSL library, which is seeded with 384-bit from /dev/random, where the seed contains 384-bits of entropy. The TOE uses this DRBG to generate all keys (as part of TLS and CSR key generation) as well as to generate salts and nonces (for password hashing and TLS respectively).

FCS_TLSC_EXT.1: The TOE's TLS client supports the ciphersuites listed in section 5.1.2.9, and allows the TLS client (used for audit export) to specify the syslog server by DNS (which uses the internal hosts file for resolution) or IP address. The TOE will use the DNS or IP as a reference identifier and check that the syslog server's certificate includes the specified identifier. The TLS client supports the Elliptic Curves Extension (specifying only P-256, P-384, and P-521) in its Client Hello, and the TOE does not require (nor allow) administrative configuration of this extension; the TOE always sends it. The TOE supports handling of wildcards within certificates, but does not support certificate/key pinning.

FCS_TLSS_EXT.1: The TOE's TLS server supports the ciphersuites listed in section 5.1.2.10, and denies versions of TLS older than TLS 1.1. The TOE uses 2048/3072-bit RSA, 2048-bit DHE and the P-256 ECDHE curve during TLS key exchange.

6.3 Identification and authentication

The Identification and authentication function is designed to satisfy the following security functional requirements:

FIA_AFL.1: The administrator can configure the number of incorrect remote authentication attempts to a value between 3 and 10 (with the default being 3), and if one exceeds that threshold, the TOE will enforce an administrator configurable (between 1 and 60 seconds) lockout of the remote administrator. The TOE does not subject the local administrator (who must directly network to the TOE and then hold the TOE's external button during power-up in order to enter configuration mode) to lockouts.

FIA_PMG_EXT.1: The TOE accepts the password characters described in section 5.1.3.2.

FIA_UAU.7: The TOE provides no feedback to the administrator during authentication other than the success or failure of their attempt.

FIA_UAU_EXT.2: The TOE provides password-based authentication of the administrator.

FIA_UIA_EXT.1: The TOE requires that the administrator first authenticate (by providing the administrative password through the TOE's TLS protected management interface) before permitting any sensitive services. However, the TOE will permit a user to perform the commands listed in section 5.1.3.5 which includes querying status information (network statistics, network metrics, output the hash of the current configuration), and (if configured) enabling override and restarting the TOE. Note that the TOE offers the administrator the ability to configure whether the TOE will allow the restart and or the override commands. As described in section 1.4.1.1, the administrator uses the same TLS management interface for all access (i.e., high and low interfaces, local and remote).

The TOE's override mode relates to the administrator specified SCADA ruleset. While normally, an administrator can only change the SCADA ruleset by loading a new configuration, the TOE allows an administrator to (optionally) specify an alternate (override) set of SCADA rules within a given configuration. Once the administrator loads the configuration, the TOE enforces the configuration's normal rules, but upon receiving a valid override command, the TOE will apply any alternate rules defined within the configuration.

FIA_X509_EXT.1/Rev: The TOE performs revocation checking of certificates during TLS client connection establishment (i.e., when the TOE acts as a TLS client connecting to a remote syslog-tls server) using CRLs that the administrator has loaded into the TOE. The TOE performs the revocation checking after having checked the validity of the server certificate and its chain. The TOE requires that TLS server certificates have the "TLS Web Server Authentication" purpose.

FIA_X509_EXT.2: The TOE relies upon the administrator to load the CA certificates (root CA, any needed intermediate certificates, as well as any CRLs needed for revocation). Because the TOE relies upon the administrator to provides all CRLs to the TOE (as often the TOE resides in a network environment where it lacks network connectivity to other networks or the Internet and thus cannot retrieve revocation information itself), if the TOE does not have any CRLs for the presented chain of certificates, the TOE accepts the server certificate as valid (not revoked). Note that if the TOE has either an incomplete set of CRLs or an expired set of CRLs, then the TOE will reject the certificate chain as invalid.

FIA_X509_EXT.3: The TOE allows an administrator to request the TOE perform on-board key generation of an RSA or ECDSA keypair and then outputs a Certificate Signing Request (CSR), that the administrator can have signed by a suitable CA.

6.4 Security management

The Security management function is designed to satisfy the following security functional requirements:

FMT_MOF.1/ManualUpdate: The TOE provides the management functions listed in section 5.1.4.1.

FMT_MOF.1/Functions: The TOE provides the management functions listed in section 5.1.4.2.

FMT_MTD.1/CoreData: The TOE only provides non-security relevant status information to Security Administrators prior to authentication.

FMT_SMF.1/FMT_SMR.2: The TOE provides a single administrative role (the Security Administrator) to administer the TOE both locally and remotely. The TOE provides the Security Administrator administrative access through its TLS server. The TOE provides the management functions described in section 5.1.4.4.

6.5 Protection of the TSF

The Protection of the TSF function is designed to satisfy the following security functional requirements:

FPT_APW_EXT.1: The TOE stores the administrative password in a salted, SHA-256 hash within an internal configuration file. Further, the TOE does not provide any way for an administrator to view, extract, or read the password. The TOE only accepts passwords during authentication attempts (or during administrative changing of the password), and calculates an eight-byte salted hash of the provided password (and then either compares it to the stored value or stores the new value depending upon whether the administrator is authenticating or changing the administrative password).

FPT_SKP_EXT.1: The TOE only possesses a private key (used for its TLS server securing administrative sessions). The TOE stores this key internally (in plaintext) and does not provide any command to access this key (and the TOE only accesses the key for use with TLSS).

FPT_STM_EXT.1: The TOE utilizes time when creating audit records and when checking syslog servers' certificates (for expiration and revocation), for session inactivity timeout, and for administrator lock out periods. The TOE obtains and maintains time by allowing the administrator to both manually specify the time as well as configure an NTP server with which the TOE synchronizes its clock.

FPT_TST_EXT.1: The TOE performs a series of power-up Known Answer Tests (a KAT for each library cryptographic algorithm that the TOE utilizes) as well as an integrity test during power-up (enumerated in section 5.1.5.4). For each self-test, the TOE uses known inputs to calculate an expected cryptographic result, and compares that result to the known result. If the calculated result matches the expected result, the test passes; if it does not match, the test fails. A failure of a power-up self-test causes the TOE to halt its boot.

FPT_TUD_EXT.1: The TOE provides an administrator the ability to view the current version of firmware (for example, on the Administration tab of Forge, one can see the 'Firmware: ' version) as well as to request that the TOE update its firmware.

As a prerequisite, the administrator first needs to obtain the new TOE firmware (in the form of a *.bau package) and then, using Forge, host that firmware on the workstation (using the Update Server tab), in order for the TOE to perform the update.

The TOE will download the new firmware, verify the digital signature on each update component, and only install the updates if the signatures verify.

6.6 TOE access

The TOE access function is designed to satisfy the following security functional requirements:

FTA_SSL.3: The administrator can configure a value between 10 and 3600 (with the default being 60) for the timeout of the administrative session in seconds. The configuration will take effect during the next administrative session.

FTA_SSL.4: The Administrator can connect and disconnect their administrative session from within the Forge software application.

FTA_SSL_EXT.1: The TOE will terminate the session after it expires.

FTA_TAB.1: The TOE provides the administrator the ability to set a banner that the TOE displays before each local and remote administrator login. Local and remote administrator logins occur through the TOE's TLS-protected management port, and the TOE will send the banner to the administrator in the same fashion (returning the configured banner through to the established TLS connection to the administrator's client application).

6.7 Trusted path/channels

The Trusted path/channels function is designed to satisfy the following security functional requirements:

FTP_ITC.1: The TOE provides the administrator the ability to transmit/export audit records to a TLS-protected syslog server. The TOE's TLS syslog configuration allows the administrator to specify the root CA certificate (as well as any needed intermediate CA certificates and any CRLs for revocation) for the TOE to use when validating the syslog server's certificate.

FTP_TRP.1/Admin: The administrator can connect to the TOE using its TLS protected administration channel.